

Appl. No 09/470,787
Amendment dated January 4, 2005

Amendments to the Claims:

This listing of claims will replace all prior version, and listings, of the claims in the application:

Listing of Claims:

1. (previously amended) A system for controlling traffic congestion within a buffered data switching network having a predetermined total buffer size, said system comprising:
- (a) a packet counter for counting a number of newly arriving packets in the switching network; and

- (b) calculation means for calculating an average queue size, \bar{Q}_t , at time t as

$$\bar{Q}_t = \bar{Q}_{t-1} \times (1 - Alpha) + Q_t \times Alpha$$

where Q_t is an instantaneous queue size, \bar{Q}_{t-1} is the average queue size at time $t-1$, and $Alpha$ is a queue-length averaging parameter,

- (c) threshold means for setting a packet-count threshold in accordance with the average queue size, for discarding a packet when the number of newly arriving packets reaches the packet-count threshold and when the average queue size exceeds a congestion threshold, and for resetting the packet counter when a packet is discarded.

2. (cancelled)

3. (previously amended) The system as in claim 1, wherein the calculation means includes means to regularly updates the average queue size using an exponential averaging technique.

4. (previously amended) The system as in claim 1, wherein $Alpha$ is assigned a value between zero and one.

5. (previously amended) The system as in claim 1, wherein a progressively increasing value of $Alpha$ is assigned with increasing level of traffic congestion.

6. (previously amended) The system as in claim 5, wherein the level of traffic congestion is indicated by the instantaneous queue size.

Appl. No 08/470,787
Amendment dated January 4, 2005

7. (previously amended) The system as in claim 1, wherein the average queue size is updated after a predetermined number of cells have arrived since a previous packet discard.
8. (previously amended) The system as in claim 1, wherein the average queue size is updated after a predetermined period of time has elapsed since a previous packet discard.
9. (previously amended) A system for controlling traffic congestion within a buffered data switching network having a predetermined total buffer size the data switching network having a queue with a queue size, said system comprising:
- (a) a packet counter for counting a number of newly arriving packets in the switching network;
 - (b) calculation means for calculating an average queue size; and
 - (c) threshold means for dividing the total queue size into a pre-selected number of N regions, for setting a packet-count threshold in accordance with the average queue size by using a descending staircase function $F(n)$, for discarding one of every $F(n)$ packets when the average queue size is in a buffer region n , $1 \leq n \leq N$ and for resetting the packet counter when a packet is discarded.
10. (previously amended) The system as in claim 9, further comprising means for detecting traffic congestion by setting a congestion threshold and comparing the average queue size with the congestion threshold, such that a congestion condition is indicated by the average queue size being equal to or above the congestion threshold, and an absence of congestion is indicated otherwise.
11. (previously amended) The system as in claim 10, wherein the packet is discarded only during the congestion condition.
12. (previously amended) The system as in claim 10, wherein the packet counter begins to operate when traffic congestion is detected, and halts operation when an absence of traffic congestion is detected.
13. (previously amended) The system as in claim 9, wherein the threshold means further includes means for dividing the total queue size into a pre-selected number of M regions, for high-priority traffic defining a high-priority congestion threshold, and the pre-selected number of N regions for low-priority traffic defining a low-priority congestion threshold, wherein the

Appl. No 09/470,787
Amendment dated January 4, 2005

threshold means sets the packet-count threshold by using two functions $F(n, m)$ and $F(m)$, such that:

when the average queue size of high-priority traffic is above the high-priority congestion threshold and is in the buffer region m , $1 \leq m \leq M$, one of every $F(m)$ high priority packets is discarded; and

when the average queue size of low-priority traffic is above the low-priority congestion threshold and is in the buffer region n , $1 \leq n \leq N$, one of every $F(n, m)$ low priority packets is discarded.

14. (previously amended) The system as in claim 13, wherein the function $F(m)$ is a descending staircase function in the buffer region m , and the function $F(n, m)$, is a multivariable function of m and n , which has a descending staircase behaviour in the buffer region n for a fixed value of m .
15. (previously amended) The system as in claim 1, further comprising means for applying a priority scheme for discarding packets, which provides a differentiated service among service classes sharing a common buffer.
16. (previously amended) The system as in claim 1, wherein the threshold means uses a look-up table.
17. (previously amended) The system as in claim 1, wherein the threshold means sets the packet-count threshold upon arrival of a new packet into the system.
18. (previously amended) The system as in claim 1, wherein the threshold means sets the packet-count threshold upon departure of a packet from the system.
19. (previously amended) A method for controlling traffic congestion within a buffered data switching network having a predetermined total buffer size, said method comprising the steps of:
 - (a) counting a number of newly arriving packets;
 - (b) calculating an average queue size wherein the average queue size at time t is calculated as:

$$\bar{Q}_t = \bar{Q}_{t-1} \times (1 - Alpha) + Q_t \times Alpha$$

Appl. No 09/470,787
Amendment dated January 4, 2005

where Q is an instantaneous queue size, \bar{Q}_{t-1} is the average queue size at time $t-1$, and $Alpha$ is a queue-length averaging parameter;

- (c) setting a packet-count threshold; and
- (d) discarding a packet and restarting the counting of newly arriving packets, when the number of newly arriving packets reaches the packet-count threshold and the average queue size exceeds a congestion threshold.

20. (cancelled)

21. (previously amended) The method as in claim 19, wherein the calculating step regularly updates the average queue size using an exponential averaging technique.

22. (previously amended) The method as in claim 19, wherein $Alpha$ is assigned a value between zero and one.

23. (previously amended) The method as in claim 19, wherein a progressively increasing value of $Alpha$ is assigned with increasing level of traffic congestion.

24. (previously amended) The method as in claim 23, wherein the level of traffic congestion is indicated by the instantaneous queue size.

25. (previously amended) The method as in claim 21, wherein the average queue size is updated after a predetermined number of cells have arrived since a previous packet discard.

26. (previously amended) The method as in claim 21, wherein the average queue size is updated after a predetermined period of time has elapsed since a previous packet discard.

27. (previously amended) A method for controlling traffic congestion within a buffered data switching network having a predetermined total buffer size, said method comprising:

- (a) counting a number of newly arriving packets;
- (b) calculating an average queue size;
- (c) dividing the predetermined total buffer size into a pre-selected number of N regions;
- (d) setting a packet-count threshold in accordance with a descending staircase function $F(n)$; and

Appl. No 09/470,787
Amendment dated January 4, 2005

(e) discarding one of every $F(n)$ packets and starting the counting of newly arriving packets, when the average queue size is in a buffer region n , $1 \leq n \leq N$.

28. (previously amended) The method as in claim 27, further comprising a step of detecting traffic congestion by setting a congestion threshold and comparing the average queue size with the congestion threshold, such that a congestion condition is indicated by the average queue size being above the congestion threshold, and an absence of congestion is indicated otherwise.

29. (previously amended) The method as in claim 28, wherein the packet is discarded only during the congestion condition.

30. (previously amended) The method as in claim 28, wherein the packet counter begins to operate when traffic congestion is detected, and halts operation when an absence of traffic congestion is detected. (delete)

31. (previously amended) The method as in claim 27, wherein the step of dividing the predetermined total buffer size includes dividing the predetermined total buffer size into both a pre-selected number of M regions, for high-priority traffic defining a high-priority congestion threshold, and a pre-selected number of N regions for low-priority traffic defining a low-priority congestion threshold, wherein the step of setting the packet-count threshold sets the packet-count threshold by using two functions $F(n, m)$ and $F(m)$, and wherein the step of discarding includes discarding one of every $F(m)$ high priority packets when the average queue size of high-priority traffic is above the high-priority congestion threshold and is in the buffer region m , $1 \leq m \leq M$, and discarding one of every $F(n, m)$ low priority packets when the average queue size of low-priority traffic is above the low-priority congestion threshold and is in the buffer region n , $1 \leq n \leq N$.

32. (previously amended) The method as in claim 31, wherein the function $F(m)$ is a descending staircase function in the buffer region m , and the function $F(n, m)$ is a multivariable function of m and n , which has a descending staircase behaviour in the buffer region n for a fixed value of m .

Appl. No 09/470,787
Amendment dated January 4, 2005

33. (previously amended) The method as in claim 19, further comprising a step of applying a priority scheme for discarding packets, which provides a differentiated service among service classes sharing a common buffer.